Computational Numerical Statistics
Assignment 3

Migla Miskinyte Reis

21.10.20

Exercise 4. (0.1 points) **Points:** ______

Let $\mathtt{X}_1, ...X_n$ be a random sample size 50, 100, 150 or 200 of some N(μ, σ^2) distribution, with $\sigma^2 = 100$. Consider the following hypothesis test on the population mean :

$$\mathtt{H}_0 : \mu \leq 500 \quad \text{vs} \quad H_1 : \mu > 500$$

for a significance level of $\alpha = 0.05$.

4.1 We will use a MC simulation to compute the empirical significance level of this test and and fill out the table of the empirical significance levels given different sample size.

```r
  m=1000; alpha=0.05; mu0=500; sigma=100; p=numeric(m) #given parameters
function_n<-function(n){  #I created function in order to use for various n
  set.seed(123)
  for(i in 1:m){
    x=x=rnorm(n,mu0,sigma) # generating the random sample
    Z=(mean(x)-mu0)/(sigma/sqrt(n)) # computing the observed value of the test statistic
    p[i]=1-pnorm(Z,0,1) # computing the p-value of the test
  }
  alphahat=mean(p<alpha) #empirical significance level
  p_value<-binom.test(alphahat*m,m,p=0.05) #testing if the computed the empirical significance level de
  return(c(alphahat,p_value$p.value)) #returning vector
}
```

One now calculates computes empirical significance and tests if it departs from $p-value$ using binomial exact test using different sample sizes.

```r
n20<-function_n(20)
n50<-function_n(50)
n100<-function_n(100)
n150<-function_n(150)
n200<-function_n(200)
m<-cbind(n20,n50,n100,n150,n200) #created a matrix of vectors for table
colnames(m) <- c("20", "50", "100", "150", "200")
rownames(m) <- c("alpha", "binom.test.p")
knitr::kable(m)
```

	20	50	100	150	200
alpha	0.0470000	0.0380000	0.0370000	0.0510000	0.0510000
binom.test.p	0.7169576	0.0817414	0.0590928	0.8844848	0.8844848

$p - value > \alpha$ for all sample sizes, we accept H_0. The test result does not depend on sample size.

4.2 Next we will use MC simulation in order to assess the power of the test and then construct a power plot to visually inspect what happens to the power of the test as we move away from H_0 μ=500 in both directions (μ 400 to 650).

```
function_n_power<-function(n){ #created a function based on different sizes
  set.seed(123)
  alpha-0.05
  power=vector()
  sep=vector()
  mu1=seq(400,650,by=10)
  m=1000
  mu0=500
  sigma=100
  for(mu in mu1){
    set.seed(789)
    p=vector()
    for(i in 1:m){
      x=rnorm(n,mu,sigma) # generating the random sample
      Z=(mean(x)-mu0)/(sigma/sqrt(n)) # computing the observed value of the test statistic
      p[i]=1-pnorm(Z,0,1) # computing the p-value of the test
    }
    power=c(power,mean(p<alpha)) # computing the empirical power of the test
    sep=power*(1-power)/m #computing std error for empirical power of hypothesis
  }
  df <- data.frame(mean=mu1,power=power,upper=power+2*sep,lower=power-2*sep) #returning dataframe that
  return(df)
}
df_n20<-function_n_power(20)
df_n50<-function_n_power(50)
df_n100<-function_n_power(100)
df_n150<-function_n_power(150)
df_n200<-function_n_power(200)
```

Now we define a function that would plot different power plots for varying sample size n.

```
power_plot<-function(mydf){

  plot(y=mydf$power, x=mydf$mean, type="b",
       xlim=c(400,650),
       ylim=c(0,1),
       xlab=expression(paste(mu)),
       ## Labels:
       ylab=expression(paste("Power")),
       main="Empirical power plots")
       abline(v=500,lty=3,col="red")

}
```

Now, we use plot all power plots side by side. We can see, that power increases when we change μ and sample size.

```
par(mfrow=c(2,3)) #2 rows and 3 columns
power_plot(df_n20)
legend('topleft', legend="n = 20", col=palette()[1:4], bty='n', cex=0.9)
power_plot(df_n50)
legend('topleft', legend="n = 50", col=palette()[1:4], bty='n', cex=0.9)
power_plot(df_n100)
legend('topleft', legend="n = 100", col=palette()[1:4], bty='n', cex=0.9)
power_plot(df_n150)
legend('topleft', legend="n = 150", col=palette()[1:4], bty='n', cex=0.9)
power_plot(df_n200)
legend('topleft', legend="n = 200", col=palette()[1:4], bty='n', cex=0.9)
```